

Šolski center Celje

Srednja šola za kemijo, elektrotehniko in računalništvo

Šolski center Celje, Srednja šola za kemijo, elektrotehniko, računalništvo

Storitev za shranjevanje datotek na oblaku

RAZISKOVALNA NALOGA



ShraniNaOblak.si

Avtorji:
Boštjan KOŠTOMAJ, R4B
Jan KUMBERGER, R4B
Franci ŠACER, R4B

Mentor:
Boštjan RESINOVIČ, mag.

Mestna občina Celje, Mladi za Celje

Celje, 2015

Šolski center Celje, Srednja šola za kemijo, elektrotehniko, računalništvo

Storitev za shranjevanje datotek na oblaku

RAZISKOVALNA NALOGA

Avtorji:
Boštjan KOŠTOMAJ, R4B
Jan KUMBERGER, R4B
Franci ŠACER, R4B

Mentor:
Boštjan RESINOVIČ, mag.

Mestna občina Celje, Mladi za Celje
Celje, 2015

»I don't need a hard disk in my computer if I can get *to the server faster… carrying around these non-connected computers is byzantine by comparison.«*

Steve Jobs, nekdanji direktor Appla

KAZALO VSEBINE

KAZALO SLIK

KAZALO TABEL IN GRAFOV

POVZETEK

V zadnjem času se pojavlja vedno več oblačnih storitev, ki nam ponujajo najrazličnejše storitve. Te storitve pa so tudi povsem spremenile filozofijo razvoja naprav, pa naj bo to pametni telefon, tablica ali osebni računalnik.

V raziskovalni nalogi smo preučili možnosti razvoja oblačne storitve shranjevanja datotek, ki bi se kosala z največjimi ponudniki tovrstnih storitev in bi zanjo porabili veliko manj sredstev, kot v njih vložijo ti giganti.

V teoretičnem delu naloge smo predstavili računalništvo v oblaku, v nadaljevanju pa smo pripravili poročilo o izdelavi naše oblačne storitve. V empiričnem delu smo predstavili rezultate našega anketnega vprašalnika.

KLJUČNE BESEDE

- računalniški oblak
- oblačna shramba
- mobilna aplikacija
- omrežni servis
- Windows Phone

1 UVOD

V zadnjem času se pojavlja vedno več oblačnih storitev, ki nam ponujajo najrazličnejše storitve. Te storitve pa so tudi povsem spremenile filozofijo razvoja naprav, pa naj bo to pametni telefon, tablica ali osebni računalnik. Te storitve nam ponujajo različne funkcije, kot so shranjevanje datotek, pošiljanje elektronske pošte, upravljanje projektnih del, sistemi za upravljanje odnosa s strankami, navidezna namizja, komunikacije, igre … Obstajajo pa tudi drugi tipi oblačnih storitev, ki so kot platforma, na primer podatkovne baze, spletni strežniki. Obstajajo pa tudi storitve, ki nam ponujajo infrastrukturo navidezne stroje, strežnike, shrambo, omrežje …

Ker pa je vedno več teh storitev, se je potrebno vprašati, kako varni so naši podatki, ki jih shranjujemo na tuje strežnike, ki jih skrbno upravljajo velike korporacije? Mar vse podatke dajejo oglaševalskim agencijam ali pa »dobronamernim« varnostnim agencijam, kot je NSA in služijo z njimi, da lahko vzpostavijo na milijone strežnikov po vsem svetu v farmah in razvijajo tajne projekte, ki spreminjajo naše človeštvo.

Po drugi strani nas tudi zanima, kakšen je strošek za vzpostavitev primerljive oblačne storitve za shranjevanje datotek, za katero bi porabili veliko manj sredstev. Zanima nas, ali bi uporabniki bili pripravljeni uporabljati novo neznano oblačno storitev, ali bi ji zaupali in shranjevali datoteke.

Zaradi vsega naštetega smo se odločili, da našo raziskavo usmerimo v poznavanje oblačnih storitev, njihove varnosti in uporabniške izkušnje. Naš cilj je tudi vzpostaviti takšno oblačno storitev za shranjevanje datotek, ki se bo kosala z največjimi ponudniki tovrstnih storitev, kot so Dropbox, OneDrive ter Google Drive.

1.1 Opredelitev problema

Temeljni cilj raziskovalne naloge je ugotoviti, kakšni so stroški pri vzpostavitvi spletne storitve za shranjevanje v oblaku in ali je z malimi stroški mogoče vzpostaviti takšno oblačno storitev za shranjevanje datotek, ki je primerljiva z ostalimi, kot so Dropbox, Google Drive, OneDrive …

1.2 Hipoteze raziskovalne naloge

Tukaj so naštete hipoteze, ki smo si jih zastavili pred začetkom raziskovanja:

1) Z zelo malimi stroški in s srednješolskim znanjem je mogoče vzpostaviti oblačno storitev za shranjevanje datotek, ki je primerljiva z Dropboxom in ostalimi.
2) Uporaba oblačnih storitev še ne presega uporabe prenosnih pomnilniških medijev.
3) Uporabniki ne zaupajo neznanim oblačnim storitvam.

1.3 Izbira raziskovalnih metod

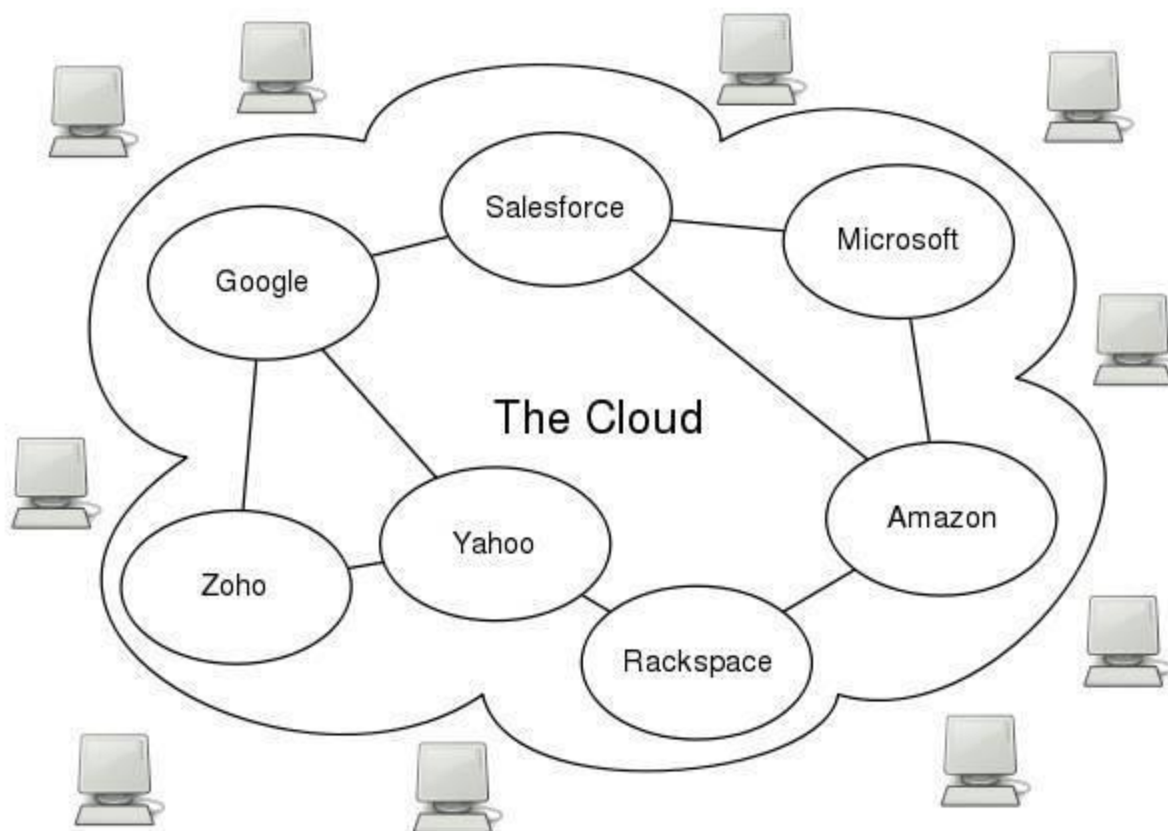
Za izhodišče raziskovalne naloge smo pripravili anketni vprašalnik, ki je priložen na koncu te naloge. Ta je bil objavljen na socialnih omrežjih in je namenjen širši javnosti.

Naslednja metoda, ki smo jo uporabili, je metoda programiranja. Pri delu smo uporabljali PHP, HTML, javascript in C# programske jezike. Pri tem smo imeli nekaj težav, saj pri rednem pouku še nismo razvijali aplikacij za Windows Phone prav tako PHP.

Kot metodo lahko navedemo tudi načrtovanje podatkovnih baz, saj je podatkovna baza jedro naše aplikacije. Pri raziskovanju smo se zanašali na naše znanje podatkovnih baz in programskih jezikov.

Predhodno smo poiskali nekaj teoretičnih izhodišč, ki neposredno ali posredno opredeljujejo značilnosti računalništva v oblaku.

2 RAČUNALNIŠTVO V OBLAKU



Slika 1 – Izraz "računalništvo v oblaku" izhaja iz shematske upodobitve Interneta, ki ga pogosto ponazorimo kot oblak (vir: Wikipedija)

O tem, kaj oblak je, obstaja več definicij, mi smo uporabljali naslednjo:
»Računalništvo v oblaku je slog računalništva, pri katerem so dinamično razširljiva in pogosto virtualizirana računalniška sredstva na voljo kot storitev preko interneta.« [2]

2.1 Storitve

Modele oblačnih storitev je ameriški nacionalni inštitut za standarde in tehnologijo (NIST) razdelil na SaaS, PaaS in IaaS [4]. V naslednjem seznamu so le-ti podrobneje opisani:

- **SaaS – Programska oprema kot storitev (angl. Software as a Service):** Programska oprema je na voljo kot storitev, dostopna preko spletnega brskalnika, brez nameščanja, vzdrževanja in nakupa licenc. Predstavlja najvišjo raven uporabe oblaka. Praviloma so to spletne aplikacije, dostopne preko interneta in prilagojene za uporabo v brskalnikih.

- **PaaS – Platforma kot storitev (angl. Platform as a Servjce):** Bistvo je nudenje platforme kot storitve oz. nudenje operacijskega sistema s pripadajočo programsko opremo. Lahko vključuje tudi orodja za razvoj aplikacij. Za vzdrževanje operacijskega sistema je zadolžen ponudnik storitev.

- **IaaS – Infrastruktura kot storitev (angl. Infrastructure as a Service):** Je najnižji nivo uporabe oblaka. Nudi nam celotno informacijsko infrastrukturo kot storitev. To pomeni, da zajema strežnike, omrežne komponente, CPU …

2.2 Tipi

Ameriški nacionalni inštitut za standarde in tehnologijo (NIST) je določil tipe oblaka kot privatni oblak, javni oblak ter mešanica obeh, hibridni oblak. [4]

2.2.1 Privatni oblak

Je rešitev, primerna za posamezno organizacijo, ki si želi imeti svoj interni oblak. Ponudnik za oblak je lahko zunanji, ali pa lahko oblak vzdržuje ter zagotavlja podjetje samo. Lastnost privatnega oblaka je, da zagotavlja dodaten nadzor nad vsem dogajanjem, ter ima povečano možnost prilagajanju podjetju samem. Podatki, shranjeni znotraj podjetja, čeprav v virtualnih strežnikih, so lahko tudi fizično dostopni. Tega javni oblak ne omogoča.

2.2.2 Javni oblak

V primeru javnega oblaka pa je vsa strojna ter programska oprema nameščena pri ponudniku, do nje dostopamo preko interneta. Podjetje mora samo plačevati ponudniku. Za servisiranje strojne opreme in za odpravljanje napak podjetje ni odgovorno. Odgovoren je ponudnik oblačne storitve.
Javni oblak je osnovni model oz. ideja računalništva v oblaku.

2.2.3 Hybrid

Je kombinacija zasebnega in javnega oblaka. Notranji viri ostanejo pod nadzorom kupca, zunanje vire pa priskrbi ponudnik storitev v oblaku. Na ta način podjetja rešijo nekatere probleme glede varnosti.

2.3 Zgodovina

Prva ideja o računalništvi v oblaku sega v leto 1960, ko John McCarthy napove, da bo računalništvo že kmalu urejeno kot javna storitev. Celoten koncept oblaka je bil raziskan leta 1966 v knjigi Douglasa Parkhilla: »The Challenge of the Computer Utility«.

V 70-ih letih so na podjetju IBM razvili operacijski sistem imenovan VM (virtual machines), ki je omogočal, da so imeli nadzorniki strežnika možnost večjih virtualnih ali navideznih strežnikov na enem fizičnem strežniku. Tako so lahko uporabniki na različnih koncih in brez stroškov uporabljali to opremo. Kljub temu pa ima vsak uporabnik svoj navidezni strežnik, s katerim upravlja le on, ostali uporabniki, ki imajo prav tako svoje navidezne strežnike, pa ne morejo dostopati na njegovega, ampak le na svojega.

Podjetja telekomunikacije, ki so prej ponujala le prenos od točke do točke (point-to-point data circuits), so v 90-ih letih začela ponujati navidezne privatne internetne povezave, ki pa so bile enake kvalitete kot prej, širše uporabljene, podjetja pa so bila zmožna ponuditi cenejše storitve. Število uporabnikov je vrtoglavo naraščalo. Ko je naraslo število uporabnikov PC-jev in interneta, je bilo logično, da se bo razvil tudi virtualni svet na internetu.

Tako je trg postal dostopen množici ljudi, ki so si prej lahko le želeli vstopiti v virtualni svet, sedaj pa je zaradi navideznih strežnikov bilo to mogoče. Podjetja so imela kar naenkrat znatno manjšo potrebo po fizični tehnologiji, ker so se začeli uporabljati navidezni strežniki. Ker je bilo uporabnikov vedno več, se je začelo pojavljati preveliko nakopičenje podatkov, kar pa ni bil tak problem, ker so bili ti fizični strežniki cenejši kot včasih. Prišli pa so do rešitve, da s programom, ki združuje fizične strežnike, lahko podatke razkropijo po več fizičnih strežnikih pa so informacije še vedno enako dosegljive. Dodajanje strežnikov k obstoječemu večjemu sistemu strežnikov sedaj ni več problem in se lahko strežniki dodajajo po potrebi. Podjetja so začela nuditi tudi "oblak" strežnike, kjer uporabnik naredi svoj navidezni strežnik, ki nastane skoraj v trenutku in na katerega lahko nalaga in si deli informacije. To vrsto strežnika lahko uporablja kdorkoli na svojem računalniku.

Pomemben del k razvoju oblaka je dodal Amazon, ki je z modernizacijo svojih podatkovnih centrov leta 2006 začel graditi popularnost oblaka.

2.4 Prednosti in slabosti

2.4.1 Prednosti

Uporabnik lahko najame storitev glede na trenutne potrebe. Uporabniku ni potrebna skrbeti za posodabljanje programske opreme, saj se to vseskozi izvršuje avtomatično. Stroški za dostop do računalniških storitev so nizki, dobra strojna oprema ni potrebna, potreben je samo spletni brskalnik.

2.4.2 Slabosti

Za dostop do programske opreme potrebujemo omrežje oz. internet. Pri počasnih internetnih povezavah se delovanje storitve poslabša ter upočasni. Vseskozi pa obstaja možnost izgube podatkov ter tudi problem varnosti le-teh.

Prednosti	**Slabosti**
- nižji stroški (plačamo samo uporabo, na pa tudi strojne ter programske opreme) - lažje vzdrževanje - višja računalniška moč - dinamična skalabilnost - hitrejši razvoj novih storitev oz. produktov - višji nivo varnosti (podatki so zbrani na centralni lokaciji ter fizično varovanje strežnikov) - dostopnost (neodvisnost lokacije uporabe, potrebujemo samo internet) - agilnost	- popolna odvisnost od omrežja - izguba nadzora (ob izpadu delovanja smo odvisni od ponudnika storitev, kdaj bo napako odpravil) - odzivnost in pasovna širina - vprašljiva prenosljivost - vprašljiva varnost - zakonodaja

Tabela 1 – prednosti in slabosti oblaka

3 POTEK DELA

3.1 Izbira programskih jezikov in razvojnih okolij

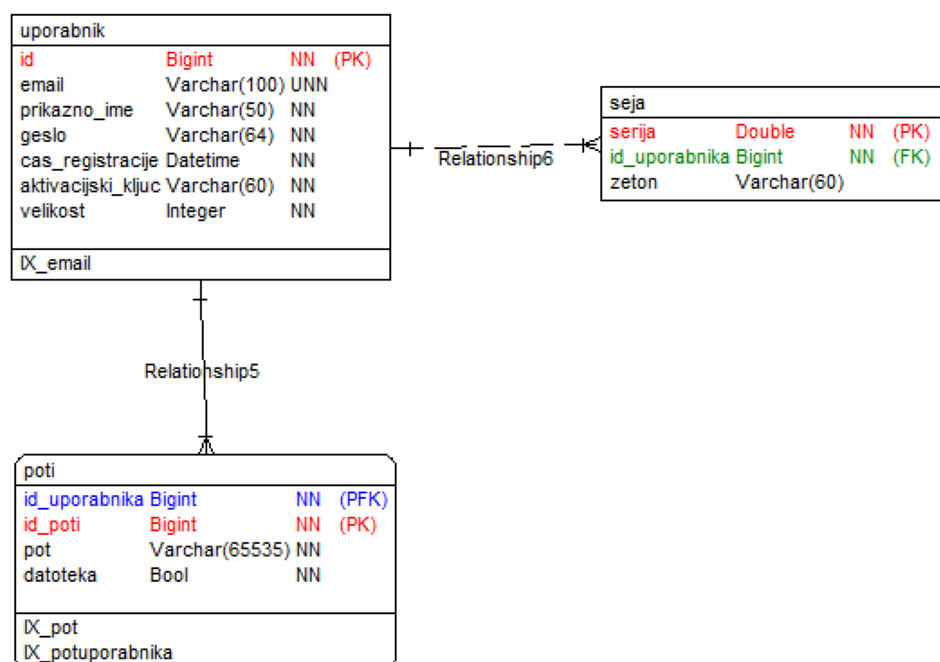
Za razvijanje spletne strani smo izbrali PHP programski jezik, saj smo imeli na voljo prost strežnik z Linuxovim okoljem in tudi zaradi enostavnosti PHP jezika. PHP aplikacijo smo razvijali v NetBeans okolju, saj ima dobro označevanje sintakse, predlaganje med pisanjem kode in tudi dobro podporo za javascript, CSS ter HTML označevalni jezik. Za podatkovno bazo smo izbrali SQL jezik in razvijali v okolju za načrtovanje podatkovnih baz CASE Studio. Za SUPB pa smo izbrali MySQL, saj z njim najlažje upravljamo v PHP jeziku. Za razvoj Windows Phone aplikacije smo izbrali C# jezik zaradi njegovega poznavanja, za razvojno okolje pa smo izbrali Visual Studio 2013.

3.2 Določitev funkcij spletnega portala

Načrtovanje aplikacij smo začeli z določanjem funkcionalnosti le-teh. Odločili smo se, da bo naša spletna aplikacija uporabnikom omogočala nalaganje in prenašanje datotek na oziroma iz strežnika, preimenovanje, premikanje datotek, ustvarjanje novih map in tudi deljenje datotek med uporabniki spletne storitve.

3.3 Načrtovanje podatkovne baze

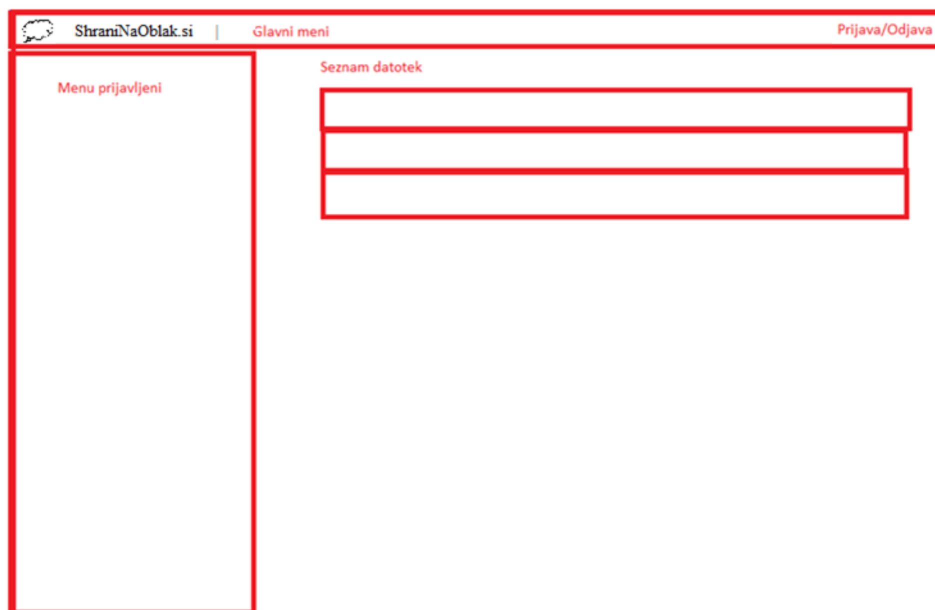
Za načrtovanje podatkovne baze smo uporabili CASE Studio, ki nam je omogočil ustvariti normaliziran načrt podatkovne baze. S tem orodjem smo imeli že veliko izkušenj, zato pri tem delu nismo imeli nobenih težav.



Slika 2 – ER diagram iz programa CASE Studio

3.4 Načrtovanje oblike spletnega portala

Obliko spletne strani smo načrtovali kar v slikarju, saj smo ga imeli vedno na voljo za idejno zasnovo naših konceptov. Potem smo stran v CSS jeziku razrezali na sekcije in dodali še ustrezne oblikovne lastnosti.



Slika 3 – Design spletnega portala

3.5 Izdelava podatkovne baze

Podatkovno bazo smo izdelali s pomočjo CASE Studia z orodjem za generiranje skript, kjer smo izbrali MySQL kot SUPB. Za boljšo učinkovitost baze je bilo pri skripti še potrebno spremeniti tip tabele, ki smo ga nastavili na InnoDB. Ta tip MySQL tabele omogoča večjo učinkovitost pri SQL transakcijah in podpira tuje ključe. Določiti je bilo potrebno tudi kodiranje tabele kot UTF-8. Potem smo SQL skripto uvozili na strežnik z MySQL Workbenchom. Pri načrtu baze smo uporabili tudi FULLTEXT indeks, da bi pospeševali iskanje po relativnih poteh datotek.

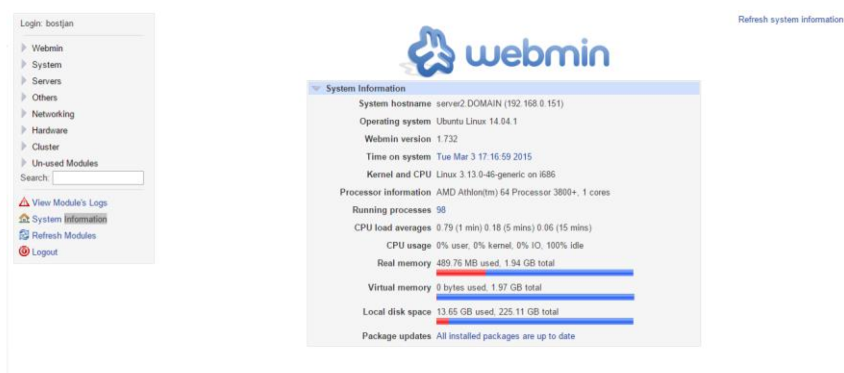
3.6 Postavitev informacijskega sistema

3.6.1 Postavitev Ubuntu strežnika

Za strežnik smo si pripravili nov računalnik, na katerega smo namestili odprtokodni operacijski sistem Ubuntu Server 14.10. Za to vrsto strežnika smo se odločili zaradi lažje namestitve in predhodnega znanja uporabe omenjenega operacijskega sistema in zelo učinkovitega delovanja. Namestitev samega strežnika nam ni predstavljala kakšnega koli problema, saj so stvari gladko tekle tako, kot smo si zamislili. Nastavili smo uporabnike za dostop do strežnika.



Slika 4 – Dodajanje in upravljanje uporabnikov



Slika 5 – Osnovna stran aplikacije Webmin

Zaradi tega, ker Ubuntu Server privzeto nima grafičnega vmesnika, smo namestili aplikacijo Webmin, ki poenostavi upravljanje in postavitev samega strežnika. Ko je bila namestitev končana, smo začeli s konfiguracijo strežnika v omrežje in vse ostale nastavitve.

Namestili smo SSH strežnik, ki se uporablja za oddaljen dostop do konzole strežnika.



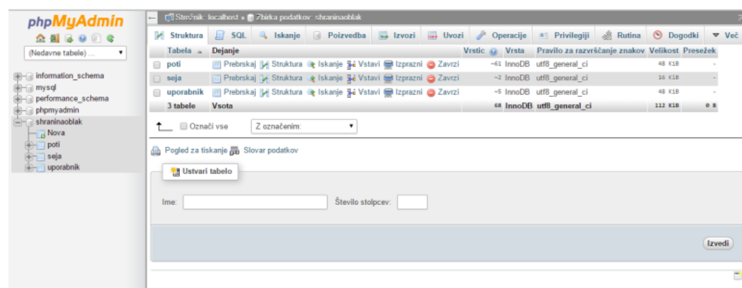
Slika 6 – Nastavitve SSH strežnika

Potem smo namestili MySQL strežnik, ki ga potrebujemo za shranjevanje podatkov in podatkov o osebah ter za delovanje spletne aplikacije. V nastavitvah MySQL strežnika smo nastavili uporabnike in njihove pravice za dostop do podatkov podatkovne baze. Zatem smo vstavili vnaprej pripravljeno skripto, ki je vsebovala programsko kodo za ustvarjanje podatkovne baze in še kreirala podatkovno bazo.



Slika 7 – MySQL strežnik

Za lažje dostopanje in upravljanje podatkovne baze smo namestili aplikacijo PHPMyAdmin, ki nam pomaga pri opravilih in popravkih podatkovne baze, pri sami varnosti in pri arhiviranju podatkov.



Slika 8 – PHPMyAdmin

Za lažje dostopanje do datotek in dodajanje datotek na strežnik smo uporabili ProFTPD strežnik, ki omogoča oddaljeno dodajanje datotek na strežnik. S tem smo omogočili dodajanje datotek v spletni strežnik Apache2.

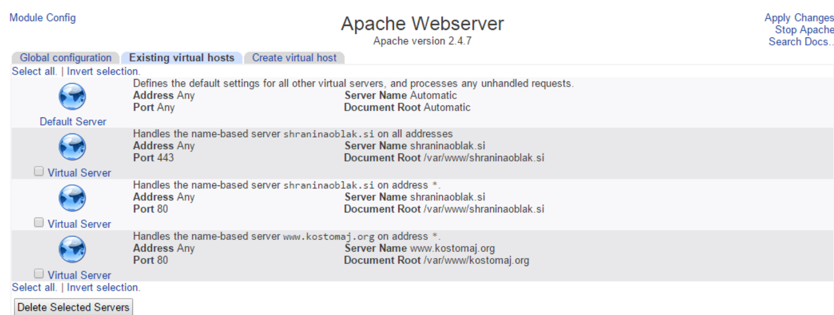


Slika 9 – ProFTPD strežnik

3.6.2 Postavitev spletnega strežnika Apache2

Na strežnik smo namestili Apache2 strežnik, ki je nujno potreben za prikazovanje spletne strani. Ko smo imeli postavljen spletni strežnik, smo zakupili spletno domeno shraninaoblak.si, ki jo uporabljamo za našo spletno aplikacijo. Ko smo imeli vse pripravljeno, smo začeli z nastavljanjem spletnega strežnika Apache2, a smo hitro naleteli na majhno težavo, saj je bila v

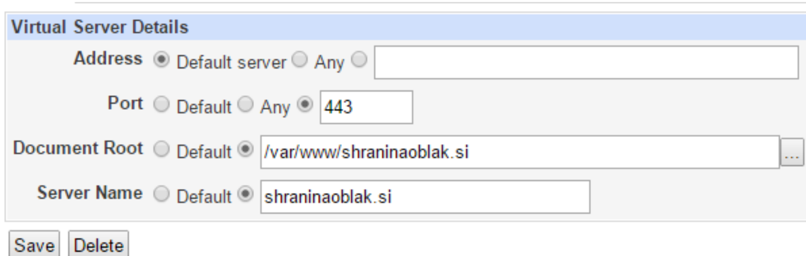
omrežju, na katerem se nahaja spletni strežnik, že nastavljena domena. To težavo smo hitro odpravili in naredili več virtualnih strežnikov, ki nam omogočajo, da lahko glede na ime, ki je napisano v brskalnik Apache, sam ugotovi, na katero spletno stran bi radi prišli in nas posledično do nje tudi pripelje.



Slika 10 – Virtualni strežniki

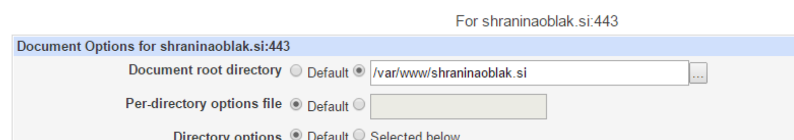
Kot vidimo na sliki imamo več virtualnih strežnikov. Prvi je privzeti, ki se odpre, če v brskalnik vpišemo samo IP strežnika. Naslednja dva virtualna strežnika sta za našo spletno stran, ampak je tisti z vrati 80 preusmerjen na virtualni strežnik z vrati 443. Vrata 443 so namenjena varni povezavi (HTTPS), za katero je potreben certifikat in je v našem primeru nujno potreben, saj preko spleta prenašamo veliko podatkov, ki pa jih drugi uporabniki na spletu ne smejo videti. S tem tudi naredimo celotno spletno stran bolj varno, da ne more priti do uhajanja podatkov med prenosom od uporabnika do strežnika, saj je naša naloga, da zagotovimo varnost našim strankam. Na zadnji virtualni strežnik je vezana domena, ki je že prej obstajala v tem omrežju.

3.6.3 Postavitev virtualnega strežnika



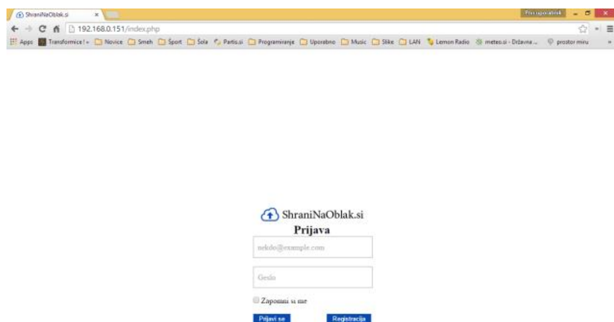
Slika 11 – Nastavitve virtualnega strežnika

Zgornja slika prikazuje, kako nastavimo virtualni strežnik. Moramo mu spremeniti vrata na 443, saj gre za varno povezavo, moramo pa nastaviti tudi ime strežnika, ki ga je najbolje nastaviti tako, kot se imenuje spletna stran, saj s tem v našem primeru rešimo razlikovanjem med dvema virtualnima strežnikoma.



Slika 12 – Nastavitev osnovne mape spletne aplikacije

Pri nadaljevanju namestitve virtualnega strežnika moramo tudi nastaviti osnovno mapo spletnega strežnika, v kateri so potem vse datoteke potrebne za prikazovanje spletne aplikacije. V nadaljevanju smo v osnovno mapo le še naložili datoteke in prikazala se je osnovna stran, kot jo vidite na sliki 4.



Slika 13 – Osnovna stran spletne strani

3.7 Izdelava spletnega portala

3.7.1 Osnovna HTML struktura

Pri strukturi smo se zanašali na uporabnikovo uporabo modernih brskalnikov, saj smo izbrali HTML5 standard. To nam je predvsem pomagalo pri validaciji vnosnih polj pri klientu. Omogočilo pa nam je tudi uporabo novejšega standarda za obliko strani CSS3.

3.7.2 CSS oblika

Pri oblikovanju spletne strani s CSS jezikom smo uporabili tudi novejše izraze za računanje širine in višine spletne strani.

3.7.3 PHP skripte

Nato smo začeli programirati PHP skripto za prikaz datotek, shranjenih na strežniku s pomočjo razvojnega okolja NetBeans IDE. Razvoj spletne aplikacije v PHP je bil kar problematičen predvsem zaradi nepopolnega poznavanja tega jezika, saj smo pluli po neznanih razburkanih vodah.

Na tej strani je bilo potrebno pridobiti podatke s strežnika, katere datoteke in mape je uporabnik shranil na strežnik. To smo dosegli s PDO gonilnikom za PHP, ki smo ga izbrali zaradi večje varnosti pred SQL vrivanjem. Pred tem smo s Query Stringom dobili zahtevo za specifični direktorij, ki smo ga potem zahtevali iz podatkovne baze. Za sortiranje smo izbrali JQuery vtičnik Table Sorter, ki nam omogoča sortiranje na strani uporabnika. Na strani smo uporabili

kar veliko javascripta za prikaz gumbov, prikaz okna za premik datotek, pošiljanje informacij strežniku in obveščanju uporabnika.

Za varnost smo poskrbeli s raznimi orodji, ki preverjajo razne ranljivosti PHP jezika. Odločili smo se tudi, da bo vsa komunikacija s strežnikom potekala prek HTTPS protokola. Certifikat smo pridobili preko zastonjskega izdelovalca certifikatov StartSSL. Za varnost prijave pa smo uporabili PHPass ogrodje za »hashing« gesel, ki ga uporabljajo številni CMS sistemi, kot je npr. WordPress.

Za varnost registracijske strani smo poskrbeli, da morajo uporabniki pred uporabo račun potrditi preko elektronske pošte, ki jo dobijo na račun in z reCAPTCHO. Pri registracijskem delu smo morali tudi pravilno nastaviti sendmail vtičnik za PHP. Za dejansko pošiljanje emailov pa smo uporabili odprtokodno knjižnico PHPMailer.

Za prenos datotek smo datoteke stisnili v ZIP datoteko, če pa je bil prenos posamezna datoteko smo jo prenesli direktno iz strežnika.

Za nalaganje datotek na strežnik smo uporabili knjižnico plupload za javascript, ki nam omogoča nalaganje datotek po kosih.

Uporabili smo tudi napredne SQL stavke z uporabo funkcij MySQL, kot so INSTR(), REPLACE(), SUBSTRING() posebej pri preimenovanju in premikanju datotek.

Za premikanje datotek med mapami smo uporabili knjižnico JQuery UI, ki nam omogoča prikaz modalnega okna s seznamom možnih map za premikanje (trenutne mape ni na seznamu).

Za skrivanje datotek pred javnostjo pa je skrbel Apache, preko katerega smo obiskovalcem onemogočili dostop do občutljivega direktorija s podatki.

3.7.4 Javascript

Pri javascriptu smo uporabljali knjižnico jQuery in vtičnika jQuery UI ter Table Sorter. Za razhroščevanje kode smo nemalokrat uporabljali tudi razvijalska orodja, ki so vgrajeni v vse moderne spletne brskalnike. Z javascriptom prav tako ni bilo težav, saj smo se osnov že naučili v šoli. Uporabili smo tudi že prej omenjeno knjižnico plupload.

Slika 14 – Izgled strani za upravljanje datotek

3.8 Načrtovanje in izdelava spletnega JSON servisa

Pri izdelavi JSON (angl. JavaScript Object Notation) servisa nismo imeli težav, saj je v PHP za JSON poskrbljeno s funkcijama json_encode() in json_decode(). Tu je primer funkcije, ki je uporabljena v spletni aplikaciji za izpis JSON objektov.

```
elseif($method==='poti')
{
    $id = is_string($_POST['id']) ? intval($_POST['id']): -1;
    $dir = is_string($_POST['dir']) ? $_POST['dir']: '';
    $email = isset($_POST['email']) ? $_POST['email']: '';
    $geslo = isset($_POST['geslo']) ? $_POST['geslo']: '';
    if(Prijava($email, $geslo)){
        header('Content-Type: application/json');
        echo json_encode(Poti($id, $dir));
    }
}
```

Slika 15 – Primer iz servisa (metoda Poti(id,dir) vrne polje nizov, ki ponazarjajo relativne poti shranjenih datotek oz. map na strežniku)

3.9 Opis funkcionalnosti spletnega portala

Ko uporabnik prvič obišče spletni portal na naslovu https://shraninaoblak.si, ga pričaka prijavni zaslon:

Slika 16 – Prijavni zaslon spletnega portala

Slika 16 – Prijavni zaslon spletnega portala

Potem uporabnik pritisne gumb »Registracija« in vpiše svoje podatke.

Slika 17 – Stran za registracijo

Ko uporabnik vnese svoje podatke, pritisne gumb »Registriraj se«.

Registracija uspešna

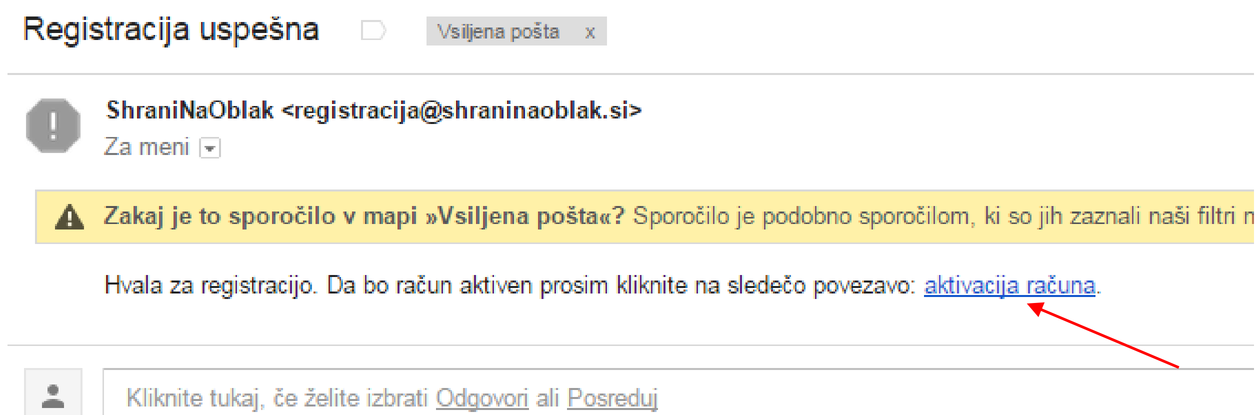
nekdo@example.com, sedaj ste registrirani. Hvala! Sporočilo je bilo poslano na email nekdo@example.com, v njem se nahaja povezava za aktivacijo računa. Sporočilo se verjetno nahaja v nezaželeni pošti!

Prijava

Slika 18 – Sporočilo o uspešni registraciji

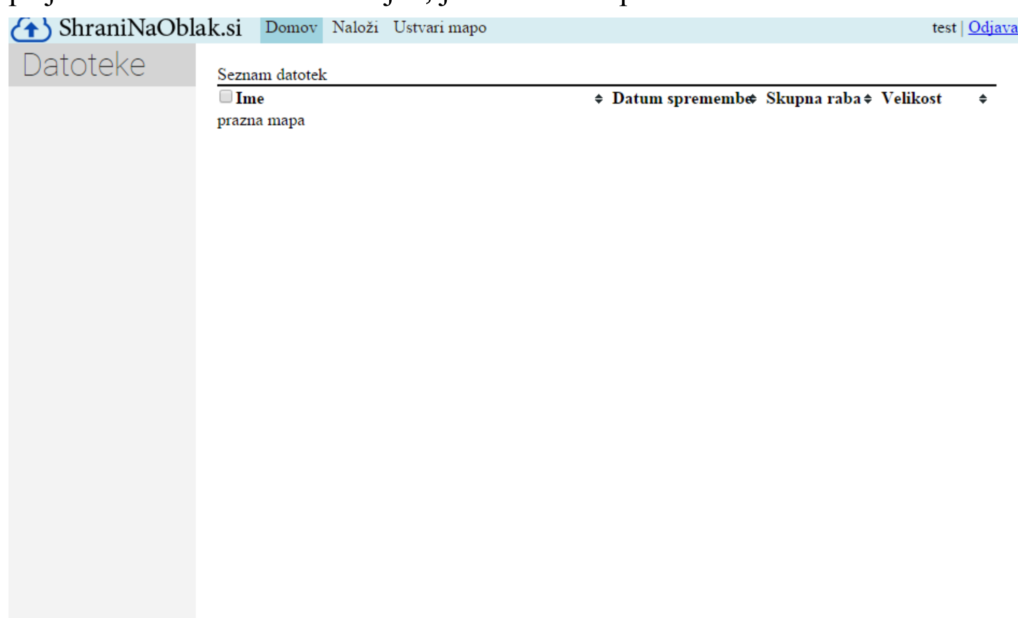
Potem uporabnik prejme potrdilo od strežnika, da mu je uspešno poslal e-pošto, kjer se nahaja aktivacijski ključ za novonastali račun. S tem uporabnik privzeto prejme 2 GB brezplačnega

prostora na strežniku. Ko uporabnik potrdi svoj račun s klikom na povezavo, ki jo je prejel v e-pošti, se lahko prijavi v aplikacijo.



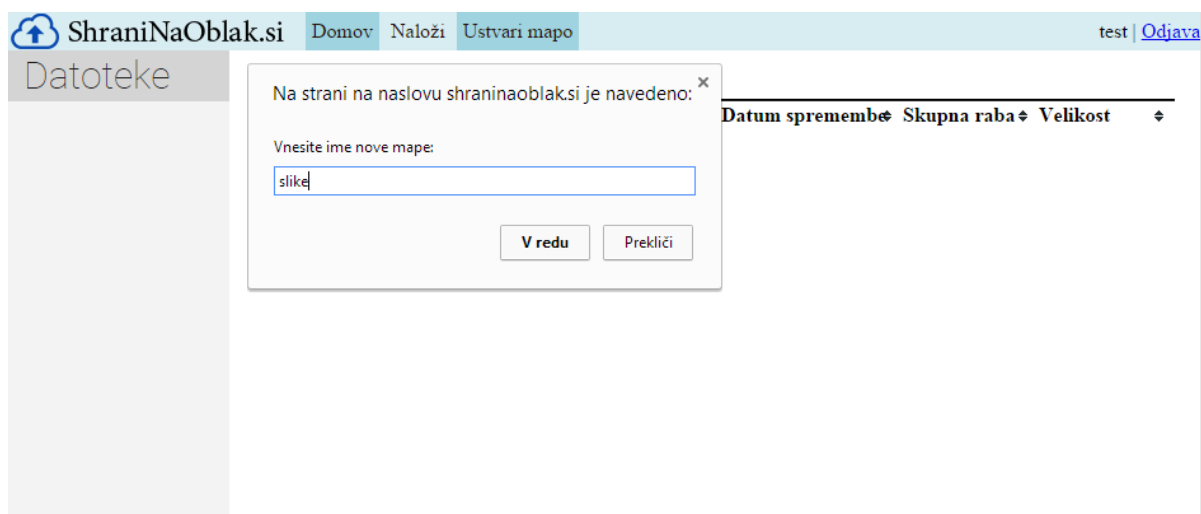
Slika 19 – Aktivacija računa in njena uspešnost

Ko je račun aktiviran in se želimo prijaviti, znova vnesemo podatke v prijavni obrazec in pritisnemo gumb »Prijavi se«. Nato se nam prikaže stran, na kateri vidimo svoje datoteke, ker pa je naš račun bil šele ustvarjen, je naš imenik prazen.

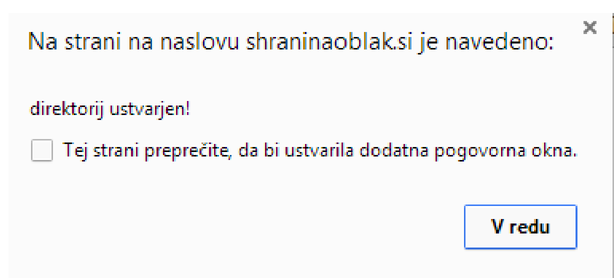


Slika 20 – Prazna mapa novonastalega računa

Da napolnimo naš imenik, lahko za začetek ustvarimo novega znotraj korenskega imenika. To naredimo tako, da v navigacijskem meniju pritisnemo »Ustvari mapo«. Nato se prikaže pozivno okno, ki nas sprašuje po imenu novega imenika.

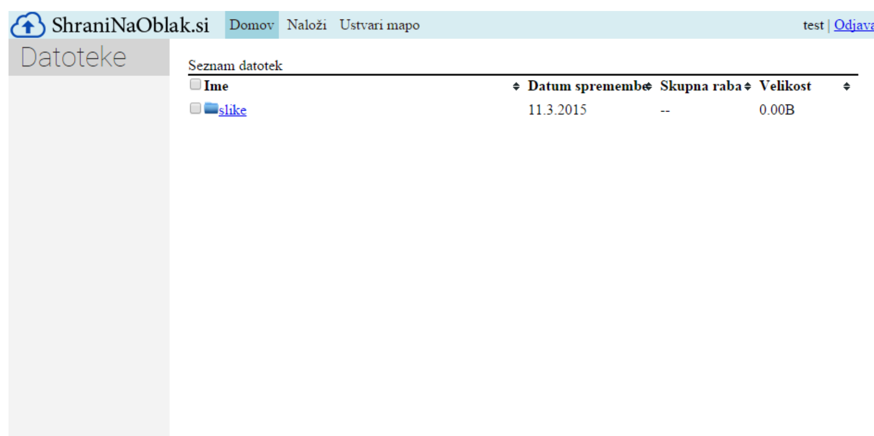


Slika 21 – Pozivno okno za ustvarjanje novega imenika



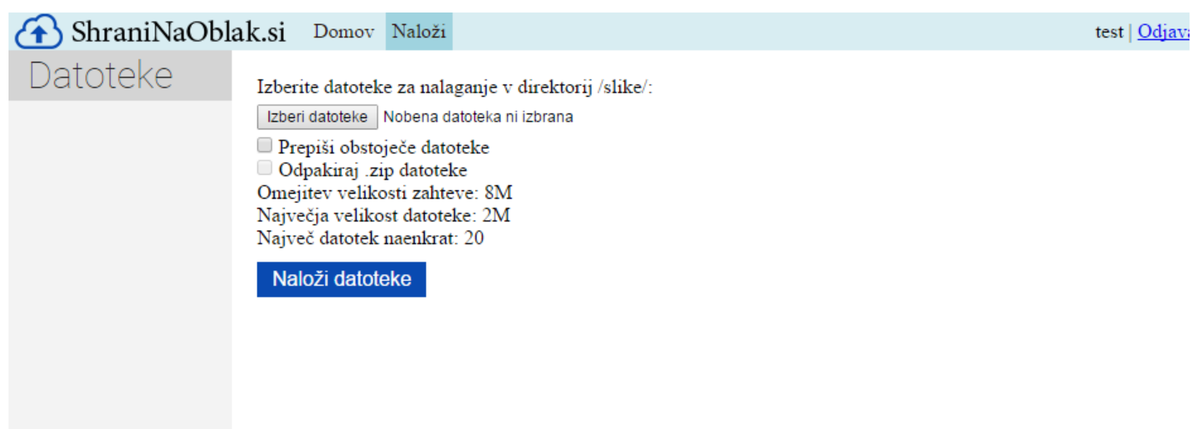
Slika 22 – Sporočilo o uspešnem kreiranju novega imenika

Ko ustvarimo imenik slike, je nov imenik viden na seznamu datotek in imenikov.



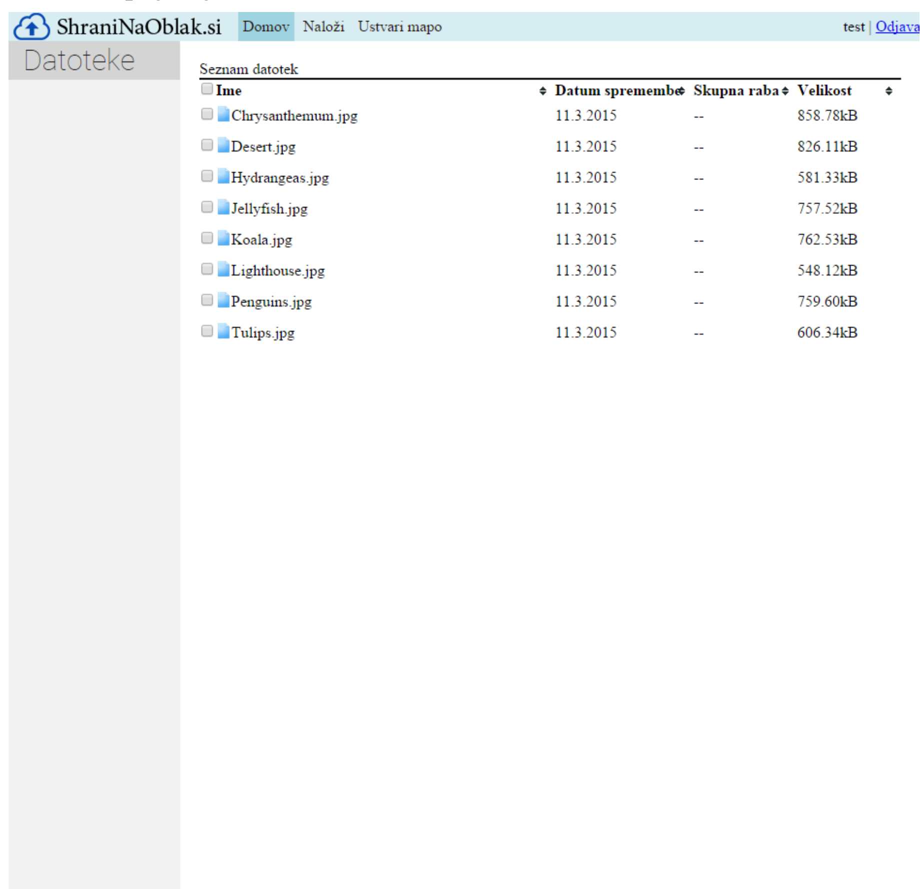
Slika 23 – Nov imenik je viden v korenski mapi

Naša naslednja naloga je naložiti naše slike v novoustvarjeni imenik. To storimo tako, da se najprej premaknemo v novoustvarjeni imenik, s klikom na ime imenika, potem pa pritisnemo gumb naloži.



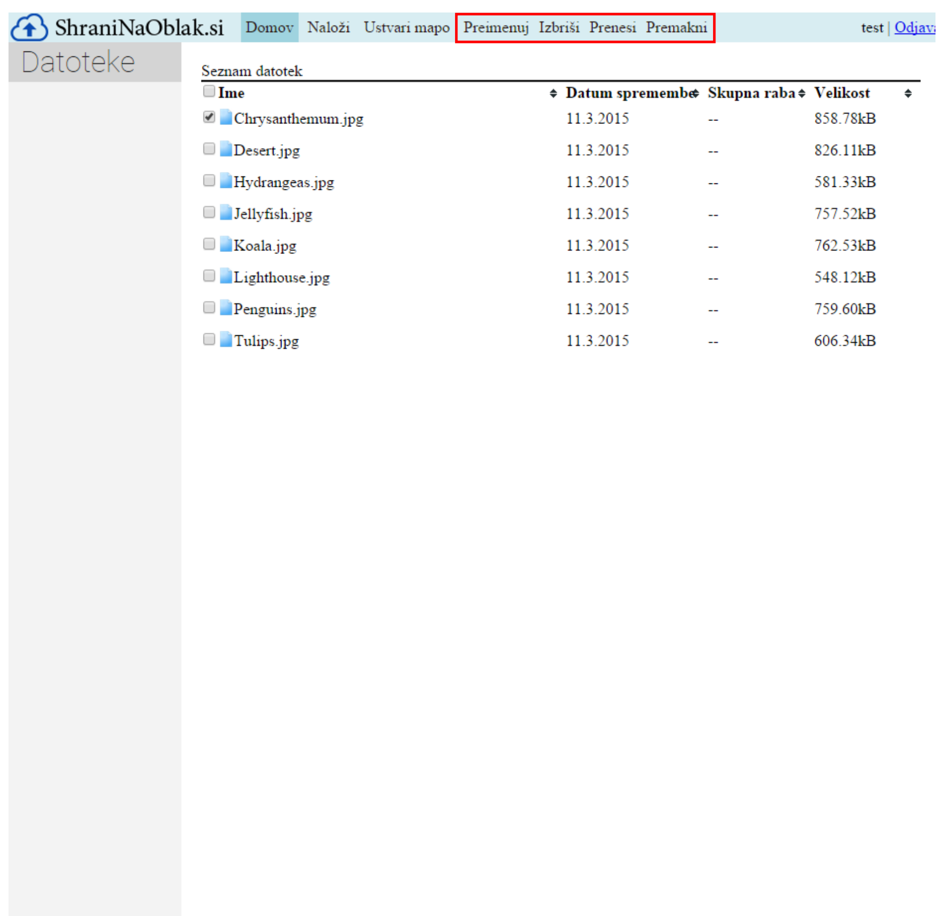
Slika 24 – Stran za nalaganje v novonastali imenik slike

Da pa datoteke naložimo, jih moramo izbrati z gumbom »Izberi datoteke« in jih poiščemo na našem računalniku. Po izboru pritisnemo na gumb »Naloži datoteke«. Po nalaganju se naše datoteke pojavijo v novonastalem imeniku.



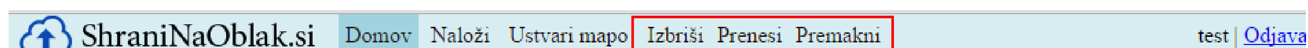
Slika 25 – Seznam naloženih datotek

Če pa določeno datoteko izberemo, jo lahko preimenujemo, izbrišemo, prenesemo ali premaknemo.



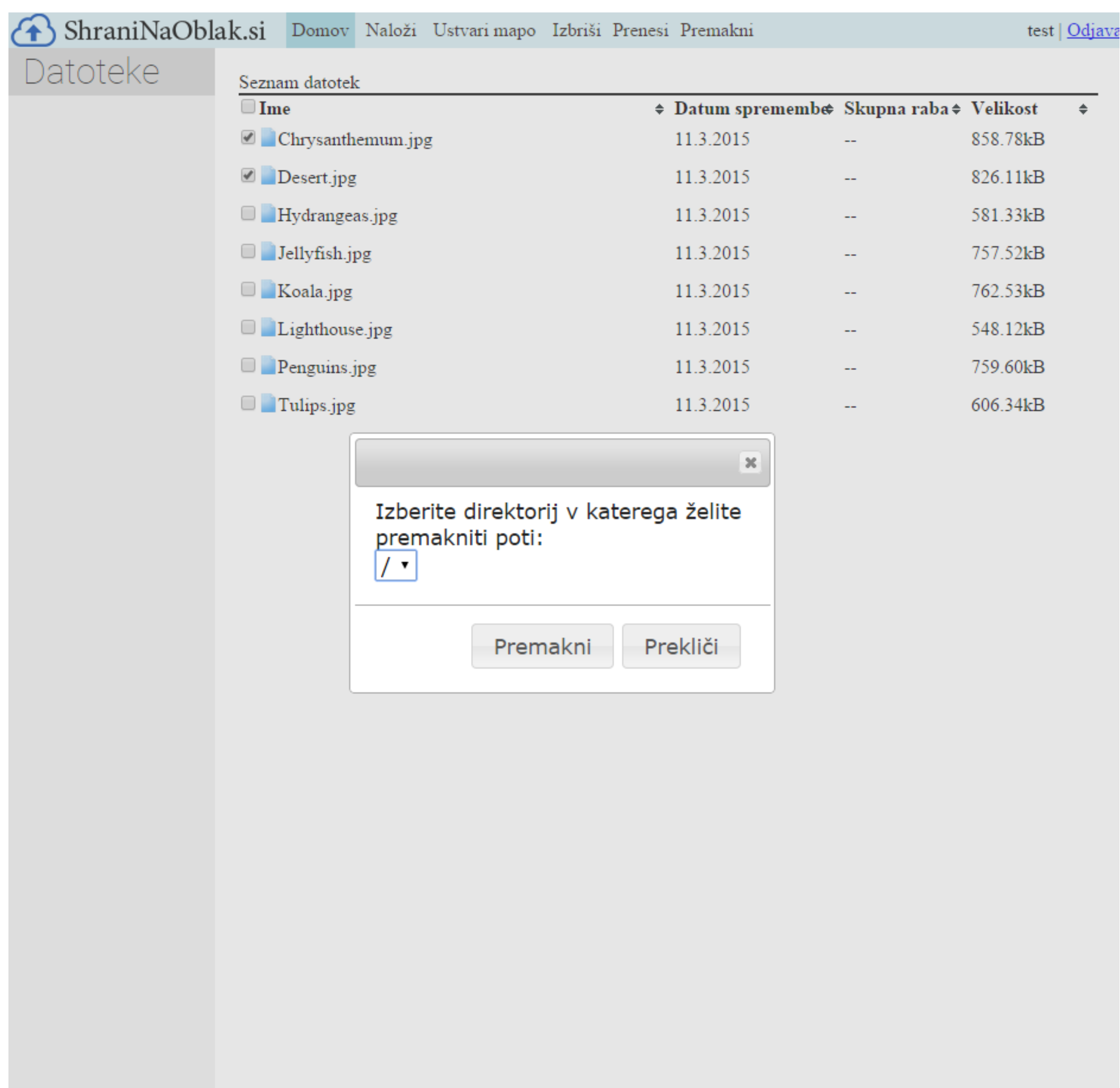
Slika 26 – Možne akcije ob izboru ene datoteke/imenika

Če izberemo 2 ali več datotek, preimenovanje ni mogoče.



Slika 27 – Možne akcije ob izboru več datotek/imenikov

Pozivno okno za preimenovanje je podobno tistemu za ustvarjanje novega imenika. Ob pritisku na navigacijski element »Izbriši« uporabnik dobi opozorilo ob brisanju. »Prenesi« ob izboru ene datoteke direktno uporabniku pošlje željeno datoteko, če pa je izbranih več datotek, jih zapakira v .ZIP datoteko in jo pošlje uporabniku. Malo poseben je gumb za premik, saj ima pojavno okno drugačno obliko, ker je bilo potrebno prikazati »dropdown« seznam imenikov, kamor je možno premakniti izbrane datoteke/imenike.



Slika 28 – Pojavno okno za izbiro destinacijskega imenika pri premikanju datotek/imenikov

Funkcionalnost za skupno rabo datotek pa še ni implementirana, a bo na voljo v končni verziji aplikacije. Imamo še kar nekaj idej za izboljšanje uporabniške izkušnje aplikacij in dodatne funkcionalnosti, ki bodo izboljšale celotno trenutno verzijo aplikacije, kot je avtomatsko razpakiranje .ZIP datotek ter »drag&drop« nalaganje, in jih bomo poskusili implementirati do predstavitve raziskovalne naloge.

3.10 Načrtovanje in izdelava Windows Phone aplikacije

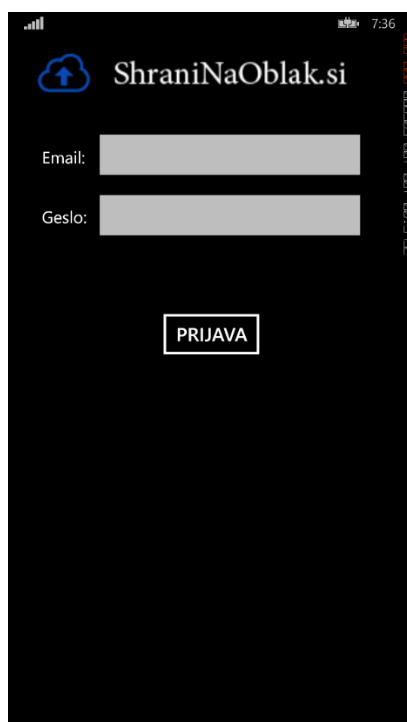
Drugi del je aplikacija za Windows Phone. Za prenos podatkov iz strežnika smo ustvarili že prej omenjeno storitev s PHP jezikom, ki prenese podatke iz baze v obliki JSON. Za varnost pri tej storitvi je ob vsaki zahtevi potrebno poslati strežniku uporabniško ime in geslo in ta komunikacija tudi poteka prek HTTPS protokola. Za dostop do JSON-a smo v C# uporabili Nuget paket Newtonsoft.JSON, ki omogoča deserializacijo JSON objektov v C# objekte. Tu smo se naučili tudi asinhronega programiranja, ki je potrebno predvsem za uporabo WebClient razreda, ki je potreben za pridobivanje podatkov iz servisa. To se v C# doseže s ključnima

besedama async in await. Ker smo za projekt ciljali verzijo Windows Phone 8, smo morali dodati tudi paket Microsoft.Bcl, ki nam omogoča dostop do funkcije WebClienta, ki namesto void vrne Task<string>, da smo lahko počakali do dokončanje prenosa strani z await. Metoda, ki čaka na Task<T>, mora tudi sama biti asinhrona. Tu smo se naučili tudi uporabe generične zbirke List, ki nam je omogočala dodajanje elementov polja, ne da bi nas morala skrbeti velikost polja. Pri oblikovanju aplikacije smo spoznali XAML označevalni jezik.

```
private void Button_Click(object sender, RoutedEventArgs e)
{
    WebClient wc = new WebClient();
    string podatki =  "email=" + email.Text + "&geslo=" + geslo.Password;
    wc.Headers[HttpRequestHeader.ContentType] = "application/x-www-form-urlencoded";
    wc.UploadStringAsync(new Uri("http://shraninaoblak.si/service.php?method=prijava"),"POST", podatki);
    wc.UploadStringCompleted += wc_UploadStringCompleted;
}

2 references
private void wc_UploadStringCompleted(object sender, UploadStringCompletedEventArgs e)
{
    bool rez = false;
    try
    {
        rez = bool.Parse(e.Result);
    }
    catch(Exception ex)
    {
        MessageBox.Show(ex.InnerException.Message);
        return;
    }
    if (rez) {
        MessageBox.Show("Prijava uspešna!", "", MessageBoxButton.OK);
        if (userSettings.Contains("email")) userSettings["email"] = email.Text;
        else userSettings.Add("email", email.Text);
        if (userSettings.Contains("geslo")) userSettings["geslo"] = geslo.Password;
        else userSettings.Add("geslo", geslo.Password);
        NavigationService.Navigate(new Uri("/SecondPage.xaml", UriKind.Relative));
    }
    else MessageBox.Show("Prijava ne uspešna!", "", MessageBoxButton.OK);
}
```

Slika 29 – Primer kode za prijavo s pomočjo servisa



Slika 30 – Prijavni zaslon aplikacije v Windows Phone



Slika 31 – Seznam datotek aplikacije v Windows Phone

Za aplikacijo smo si želeli, da bi ponujala enake funkcionalnosti kot spletni portal, vendar se bo s tem verjetno pojavilo nekaj težav. Zagotovili bomo vsaj, da bo iz aplikacije mogoče prenašati in nalagati datoteke iz/na strežnik. Aplikacija pa bo namreč dokončana šele do predstavitve te naloge. Prejšnji 2 sliki sta osnutek za aplikacijo.

4 ANALIZA ANKETE

V tem delu vam želimo predstaviti rezultate anketnega vprašalnika, ki ga najdete v prilogi. Anketirali smo 73 ljudi iz socialnih omrežij, ki so si vzeli čas, da izpolnijo naš anketni vprašalnik. Vprašalnik je bil objavljen preko Google Obrazcev. Spraševali smo jih o uporabi oblačnih storitev, kaj pričakujejo od njih in kateri medij uporabljajo za shranjevanje podatkov.

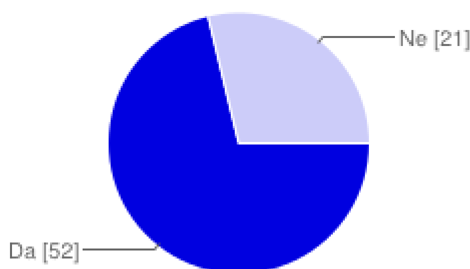
Pri sestavljanju anketnega vprašalnika smo imeli nemalo težav, saj so se vedno znova porajala nova vprašanja, iz katerih bi izvedeli tudi, ali so uporabniki pripravljeni sprejeti tudi kakšno novo oblačno storitev, ki ni vezana na kakšno večjo korporacijo.

4.1 Predstavitev rezultatov anketnega vprašalnika

1. **Ali zaupate oblačnim storitvam?**

odgovor	št.	delež
Da	52	71 %
Ne	21	29 %

Tabela 1 – prikazuje zaupanje v oblačne storitve



Graf 1 – prikazuje zaupanje v oblačne storitve

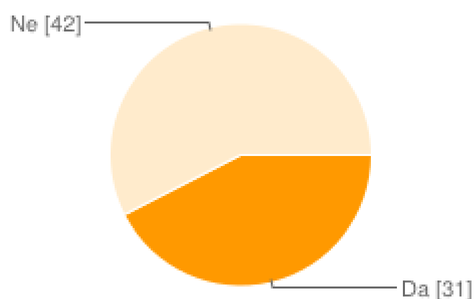
REZULTATI

Kot je razvidno iz tabele in grafa, večina anketirancev zaupa oblačnim storitvam. Ta rezultat ni presenetljiv, saj se uporabniki takih storitev po navadi ne zavedamo varnostnih vprašanj, ki se pojavijo pri uporabi oblačnih storitev.

2. **Ali bi uporabljali novo oblačno storitev, ki jo je razvilo novo, še neznano podjetje?**

odgovor	št.	delež
Da	31	42 %
Ne	42	58 %

Tabela 2 – prikazuje odprtost za storitve, ki jih razvije novo podjetje



Graf 2 – prikazuje odprtost za storitve, ki jih razvije novo podjetje

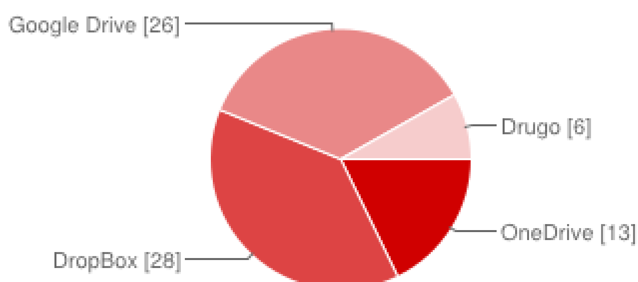
REZULTATI

Kot je razvidno iz grafa in tabele tukaj odgovor nima večine, se pa nagiba k nezaupljivosti za nove storitve. Nezaupljivost je kar predvidljiva, saj je res težko zaupati podatke nekemu neznanemu podjetju.

3. Katera oblačna storitev vam je najbolj všeč?

odgovor	št.	delež
OneDrive	13	18 %
DropBox	28	38 %
Google Drive	26	36 %
Drugo	6	8 %

Tabela 3 – prikazuje priljubljenost določenih oblačnih storitev za shranjevanje datotek



Graf 3 – prikazuje priljubljenost določenih oblačnih storitev za shranjevanje datotek

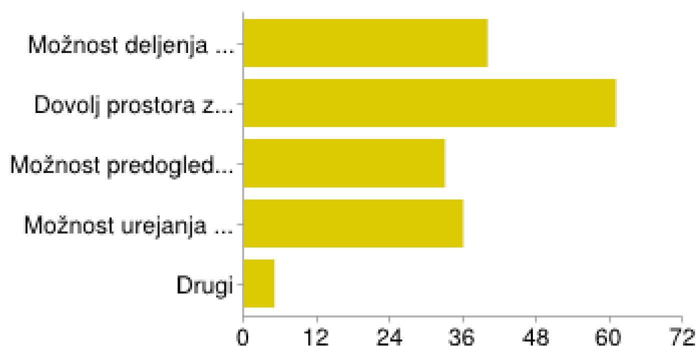
REZULTATI

Kot je razvidno iz tabele in grafa, je najbolj priljubljena storitev za shranjevanje datotek Dropbox. Tak rezultat je pričakovati, saj je bil Dropbox eden izmed prvih, ki so ponudili takšno storitev, ki je za 2 GB prostora zastonj in enostaven za uporabo.

4. Kaj pričakujete od oblačne storitve, ki ponuja možnost shranjevanja datotek?

odgovor	št.	delež
Možnost deljenja datotek	40	55 %
Dovolj prostora za shranjevanje	61	84 %
Možnost predogleda datotek	33	45 %
Možnost urejanja datotek	36	49 %
Drugi odgovori	5	7 %

Tabela 4 – prikazuje pričakovane funkcionalnosti oblačne storitve za shranjevanje datotek



Graf 4 – prikazuje pričakovane funkcionalnosti oblačne storitve za shranjevanje datotek

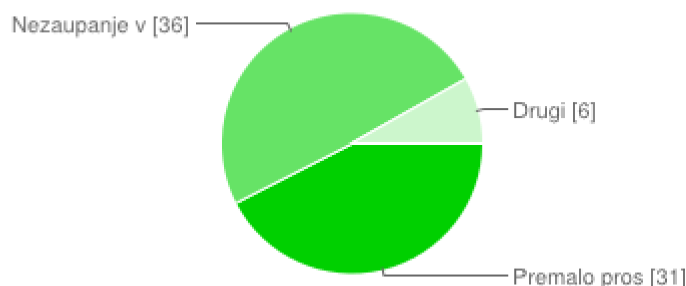
REZULTATI

Pri tem vprašanju je bila možna izbira več odgovorov. Kot je razvidno iz tabele in grafa, pa skoraj vsi želijo imeti dovolj prostora za shranjevanje. Tak rezultat je bil sicer pričakovan, ampak si nismo mislili, da bo prevladal prostor, saj vedno več storitev ponuja večje pakete že po 50 GB brezplačno, kar je veliko glede na povprečno hitrost internetnih povezav v Sloveniji.

5. Kaj vas moti pri oblačnih storitvah?

odgovor	št.	delež
Premalo prostora	31	42 %
Nezaupanje v varnost podatkov	36	49 %
Drugi odgovori	6	8 %

Tabela 5 – prikazuje, kaj moti uporabnike oblačnih storitev za shranjevanje datotek



Graf 5 – prikazuje, kaj moti uporabnike oblačnih storitev za shranjevanje datotek

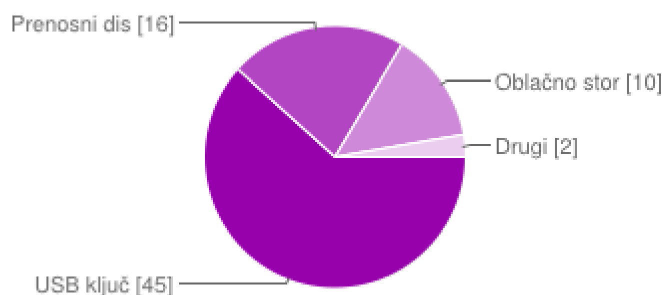
REZULTATI

Kot je razvidno iz tabele in grafa, uporabnike najbolj moti njihovo nezaupanje v varnost oblačnih storitev. To je za pričakovati, saj se zaradi raznih afer, kot je tista z iCloudom, res vprašamo, ali so naši podatki in datoteke varni.

6. **Kateri medij najbolj uporabljate za shranjevanje podatkov?**

odgovor	št.	delež
USB ključ	45	62 %
Prenosni disk	16	22 %
Oblačno storitev	10	14 %
Drugi odgovori	2	3 %

Tabela 6 – prikazuje najbolj uporabljen medij za shranjevanje podatkov



Graf 6 – prikazuje najbolj uporabljen medij za shranjevanje podatkov

REZULTATI

Kot je razvidno iz tabele in grafa, je prepričljivo najbolj uporabljen medij USB ključek. To smo pričakovali, saj tudi sami v večini uporabljamo USB ključ, saj je zanesljiv, ko ni internetne povezave.

7. **Kolikšna se vam zdi primerna cena za zakup prostora v oblaku?**

REZULTATI

Tu je bilo kar nekaj odgovorov zastonj, kar veliko jih ni zraven napisalo količine prostora in za koliko časa. Ostali so bili pripravljeni povprečno plačati 10 € mesečno za terabajt prostora.

5 ZAKLJUČEK

V tem delu predstavljamo zaključke v zvezi z raziskavo. Na splošno pa smo zadovoljni z rezultati raziskovalne naloge in na novo pridobljenim znanjem.

5.1 Potrditev hipotez

Vse hipoteze so bile potrjene:

1) Z zelo malimi stroški in s srednješolskim znanjem, ki ga je bilo potrebno nadgraditi, smo vzpostavili oblačno storitev za shranjevanje datotek, ki je primerljiva z Dropboxom in ostalimi, saj omogoča večino funkcij, kot jih ponujajo ostale storitve. Pod stroške lahko štejemo registracijo domene, ki je okoli 10 €/leto, sem pa je potrebno prišteti še stroške za strežnik (mi smo ga imeli že na voljo, v realnem primeru bi bilo storitev bolje postaviti na oblak, kot je Windows Azure za že od 10 €/mesec na virtualnem stroju), ki je lahko skorajda katerikoli moderen osebni računalnik, in pa stroške za internetne storitve, ki za povprečno internetno povezavo znašajo nekaj nad 30 €/mesec.
2) Z anketnim vprašalnikom smo ugotovili, da so ključki zdaleč najbolj uporabljen medij.
3) Z anketnim vprašalnikom smo ugotovili, da jih večini ni pripravljenih uporabljati oblačno storitev neznanih podjetij, saj jim ne zaupajo svojih podatkov. Kljub temu da so uporabniki zelo nezaupljivi, bi se skušali prebiti najprej pri sošolcih in ti bi povedali naprej svojim prijateljem in tako dalje. Zanašali bi se torej na prenos iz ust do ust in kasneje še z oglaševanjem.

5.2 Težave ob izvedbi

Pri izdelavi smo naleteli predvsem na nove programerske težave, ki smo jih z željo dokončanja projekta uspešno rešili. Pomagali smo si z raznimi vodiči za programiranje. Največ težav se je pojavilo pri izdelavi servisa, saj nismo vedeli, kje začeti, ker še nismo izdelovali servisov, vendar smo težave na koncu rešili.

5.3 Kaj smo se naučili?

Kot smo že navedli v rezultatu 1. hipoteze, je bilo naše znanje potrebno nadgraditi, sicer pa smo že imeli veliko potrebnega predznanja. Spoznali smo določene novosti PHP jezika, predvsem o varnosti. Bolje smo spoznali tudi razvojno okolje NetBeans. Naučili smo se tudi postaviti Ubuntu Server z vsemi dodatki. Na novo pa smo spoznali razvijanje mobilnih aplikacij za platformo Windows Phone. Naučili smo se tudi nekaj več o timskem delu, ki je bil kar pomemben del naše naloge.

5.4 Smernice za nadaljnje delo

Kot nadaljnje delo bi lahko šteli razvoj mobilnih aplikacij za ostale platforme in dodajanje funkcionalnosti, kot je pregledovanje ter urejanje datotek brez prenosa datotek iz strežnika. Glede spletne aplikacije bi se dalo izboljšati uporabniški vmesnik tako, da bi podpiral mobilne naprave in odzivno obliko (angl. responsive web design). Izven raziskovalne naloge bi lahko implementirali tudi sistem za plačevanje nadgradenj prostora na strežniku za uporabnika.

6 VIRI IN LITERATURA

1. **Wikipedia** *Cloud computing*. Dostopno na spletnem naslovu: http://en.wikipedia.org/wiki/Cloud_computing (Zadnji dostop: 22. 2. 2015).
2. **Wikipedija** *Računalništvo v oblaku*. Dostopno na spletnem naslovu: http://sl.wikipedia.org/wiki/Ra%C4%8Dunalni%C5%A1tvo_v_oblaku (Zadnji dostop: 21. 2. 2015).
3. **Brišnik**, T. in **Šinkar**, N. *Gručenje in oblak v računalništvu*. Raziskovalna naloga. Velenje: Šolski center Velenje, 2011.
4. **W3Schools** *Spletna stran za pomoč pri programskih jezikih*. Dostopno na spletnem naslovu: http://www.w3schools.com/ (Zadnji dostop: 20. 2. 2015).
5. **NIST** *The NIST Definition of Cloud Computing*. Dostopno na spletnem naslovu: http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf (Zadnji dostop: 25. 2. 2015).
6. **PHP Group** *Uradna dokumentacija PHP jezika*. Dostopno na spletnem naslovu: http://php.net/manual/en/ (Zadnji dostop: 26. 2. 2015).
7. **Microsoft MSDN** *Vodila za razvoj Windows Phone aplikacij*. Dostopno na spletnem naslovu: https://msdn.microsoft.com/library/windows/apps/ff402535.aspx (Zadnji dostop: 28. 2. 2015).

PRILOGE

Anketni vprašalnik

Anketa

Smo dijaki 4. letnika in raziskujemo uporabo ter razširjenost oblačnih storitev.

* Zahtevano

Ali zaupate oblačnim storitvam? *

- ◯ Da
- ◯ Ne

Ali bi uporabljali novo oblačno storitev, ki jo je razvilo novo, še neznano podjetje? *

- ◯ Da
- ◯ Ne

Katera oblačna storitev vam je najbolj všeč? *

- ◯ OneDrive
- ◯ DropBox
- ◯ Google Drive
- ◯ Drugo

Kaj pričakujete od oblačne storitve, ki ponuja možnost shranjevanja datotek? *

- ☐ Možnost deljenja datotek
- ☐ Dovolj prostora za shranjevanje
- ☐ Možnost predogleda datotek
- ☐ Možnost urejanja datotek
- ☐ Drugo: ____

Kaj vas moti pri oblačnih storitvah? *

- ◯ Premalo prostora
- ◯ Nezaupanje v varnost podatkov
- ◯ Drugo: ____

Kateri medij najbolj uporabljate za shranjevanje podatkov? *

- ◯ USB ključ
- ◯ Prenosni disk
- ◯ Oblačno storitev
- ◯ Drugo: ____

Kolikšna se vam zdi primerna cena za zakup prostora v oblaku? *

____

Pošlji

Nikoli ne pošiljajte gesel prek Google Obrazcev.